

# 学生面试问题

## 一、 问题重述

高校自主招生是高考改革中的一项新生事物，现在仍处于探索阶段。某高校拟在全面衡量考生的高中学习成绩及综合表现后再采用专家面试的方式决定录取与否。该校在今年自主招生中，经过初选合格进入面试的考生有  $N$  人，拟聘请老师  $M$  人。每位学生要分别接受 4 位老师（简称该学生的“面试组”）的单独面试。面试时，各位老师独立地对考生提问并根据其回答问题的情况给出评分。由于这是一项主观性很强的评价工作，老师的专业可能不同，他们的提问内容、提问方式以及评分习惯也会有较大差异，因此面试同一位考生的“面试组”的具体组成不同会对录取结果产生一定影响。为了保证面试工作的公平性，组织者提出如下要求：

- Y1. 每位老师面试的学生数量应尽量均衡；
- Y2. 面试不同考生的“面试组”成员不能完全相同；
- Y3. 两个考生的“面试组”中有两位或三位老师相同的情形尽量少的；
- Y4. 被任意两位老师面试的两个学生集合中出现相同学生的人数尽量少。

请回答如下问题：

问题一：设考生数  $N$  已知，在满足 Y2 条件下，说明聘请老师数  $M$  至少分别应为多大，才能做到任两位学生的“面试组”都没有两位以及三位面试老师相同的情形。

问题二：请根据 Y1~Y4 的要求建立学生与面试老师之间合理的分配模型，并就  $N=379$ ,  $M=24$  的情形给出具体的分配方案（每位老师面试哪些学生）及该方案满足 Y1~Y4 这些要求的情况。

问题三：假设面试老师中理科与文科的老师各占一半，并且要求每位学生接受两位文科与两位理科老师的面试，请在此假设下分别回答问题一与问题二。

问题四：请讨论考生与面试老师之间分配的均匀性和面试公平性的关系。为了保证面试的公平性，除了组织者提出的要求外，你们认为还有哪些重要因素需要考虑，试给出新的分配方案或建议。

## 二、 符号说明

$x_{ij}$  :  $x_{ij}$  取 1 表示第  $i$  个学生被第  $j$  位教师面试，0 则表示没有；

$ST_{N \times M}$  :  $M$  位老师面试  $N$  个学生的分配矩阵；

$$ST_{N \times M} = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,M} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,M} \\ \cdots & \cdots & \cdots & \cdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,M} \end{bmatrix}$$

$S_i$  :  $ST_{N \times M}$  矩阵中第  $i$  个行向量；

$T_j$  :  $ST_{N \times M}$  矩阵中第  $j$  个行向量；

$F(S, n)$  : 向量  $S$  中元素值大于  $n$  的元素个数；

$[n]$  :  $n$  值向上取整；

## 三、 问题的分析以及模型的建立、求解

在忽略各子问题的约束和优化条件基础上，此学生面试问题在形式上属于标准的 0/1 整数规划问题，基本模型建立如下：

$$ST_{N \times M} = \begin{bmatrix} x_{1,1} & x_{1,2} & \dots & x_{1,M} \\ x_{2,1} & x_{2,2} & \dots & x_{2,M} \\ \dots & \dots & \dots & \dots \\ x_{n,1} & x_{n,2} & \dots & x_{n,M} \end{bmatrix}$$

其中  $x_{i,j}$  为 0 或 1；

$$\text{目标函数 } \max Z = \text{countzero} \left( \sum_{i=1}^n S_i \right) \quad (1)$$

$$\text{满足约束条件 } \sum_{j=1}^{4n} x_{i,j} = 4 \quad (i = 1, 2, \dots, n) \quad (2)$$

其中函数  $\text{countzero}(s)$  计算向量  $S$  中零元素的数目。

## ◆ 问题 1

### 1. 问题分析：

根据问题描述，此问题由于约束条件的不同，可以分为如下两个子问题：

问题 1.1：任两位学生的“面试组”都没有两位面试老师相同；

其数学表达如下：

$$F(S_i + S_j, 2) < 2 \text{ 其中 } i, j = 1, \dots, n \text{ 且 } i < j \quad (3)$$

问题 1.2：任两位学生的“面试组”都没有三位面试老师相同；

其数学表达如下：

$$F(S_i + S_j, 2) < 3 \text{ 其中 } i, j = 1, \dots, n \text{ 且 } i < j \quad (4)$$

综合上述模型和约束条件，本问题可以采用 0/1 整数规划的方法求解。

考虑实际的求解环境，为了在求解精度和求解性能上达到一定的平衡，以下将用几种不同的模型求解。

### 2. 模型求解

#### ■ 问题 1.1

##### 1) 0/1 整数规划方式求解

采用 0/1 整数规划求解该模型的算法见附件中算法 1。

该算法可以求得确定学生数  $N$  基础上，最少需要的老师数  $M$ ，以及其对应的分配矩阵，模型求得的结果为问题可行域上的最优解。

但是当  $N$  较大时，可行域空间极为庞大，搜索性能较差。为了进一步改善求解性能，本文进一步提出其他模型进行改进。

##### 2) 组合数方式求解

考虑到问题要求求得老师数  $M$  至少为多大，即需要确定一个  $M$  的下限取值。为此用组合数方法分析满足问题条件的解的性质：

性质 1：可行解  $ST_{N \times M}$  每一行 4 个元素为 1 的（列）数位可以组成  $C_4^4$  个数位对；各行生成的  $C_4^4$  个（列）数位对组成数量为  $NC_4^4$  的数位对集合，在此集合中任何两个数位对都不相同。

例：假定第  $i$  行 4 个元素 1 分别位于第  $a, b, c, d$  列，则第  $i$  行的 4 个元素为 1 的数位可以组成  $(a, b), (a, c), (a, d), (b, c), (b, d), (c, d)$  6 个数位队。

证明：性质 1 中前一部分显而易见。后一部分的证明可以采用反证法：

假定所有的数对中存在 2 个数对相同，则由于同一行的 4 个数中不可能产生相同的两个数队，故这两个数对必然产生于不同的行  $i$  和  $j$ 。由此可以推断第  $i$  行和第  $j$  行存在两个列数位相同的元素 1，即两个学生有两位面试老师相同，和题目约束条件矛盾。因此性质得证。

性质 2：可行解  $ST_{N \times M}$  满足  $NC_4^2 \leq C_M^2$

证明：由性质 1 可得，按照性质 1 的列数位对生成规则，所有行可以生成  $NC_4^2$  个两两不同的数位对。由于所有的数位对都是基于 1 到  $M$  的列号产生的二元组合  $(i, j) (i=1, \dots, m; j=1, \dots, m; i \neq j)$ ，故其数目不会超过 1 到  $M$  能够产生的二元组合最大数目  $C_M^2$ 。故性质 2 得证。

由性质 2 可以推出  $M$  的取值下界条件：

$$M \geq \frac{1 + \sqrt{1 + 48N}}{2} \quad (5)$$

即结论： $M$  至少应为  $\left\lceil \frac{1 + \sqrt{1 + 48N}}{2} \right\rceil$  时，才可能满足任两位学生的“面试组”都没有两位面试老师相同。

由于上述计算过程只考虑了该问题的必要性条件，因此求出的下界不一定是问题的下确界，更为精确的下确界模型为  $NC_4^2 \leq C_M^2 - t$ ，其中  $t$  为关于  $N$  和  $M$  的函数。实际的模型计算较为复杂，故此本文接下来将通过其他方法寻求更为精确的模型，并在一定程度上通过其对本模型进行验证。

### 3) 基于最短路径的阶段递进搜索算法

- ✓ 假定 1：存在  $N$  个学生对应  $M$  位老师的最优分配方案之一的情况下，该最优分配方案满足各老师的面试负荷量相对均衡；
- ✓ 假定 2：存在满足假定 1 的  $ST_{(N+1) \times M}$  的最优分配方案，该方案可由某一个满足假定 1 的  $ST_{N \times M}$  的最优分配方案在不改变现有分配的基础上扩展取得（即  $ST_{(N+1) \times M}$  中存在一个  $N \times M$  子矩阵等于  $ST_{N \times M}$ ）；

由假定 1 和假定 2 可以得出以下两条搜索规则：

规则 1：从  $ST_{N \times M}$  的某一最优分配方案向  $ST_{(N+1) \times M}$  的某一最优方案扩展时，只需在  $ST_{N \times M}$  的基础上增加第  $N+1$  行，并调整其中非零元素的位置（同时方案根据需要增加一定数量的列或者保持列数不变）；

规则 2：在对  $N+1$  个学生进行老师分配时，总是先选取在  $N$  个学生数下所有老师中面试任务最轻的第一位老师；

由规则 1 及规则 2 编写的基于最短路径的阶段递进搜索算法见附件算法 2。该算法产生的一系列  $(N, M)$  数对结果（见附件数据 1）可应用于对上述各模型的检验及模型拟合度分析。

### 4) 模型验证及比较

根据附件中的算法 2 得到一批  $(N, M)$  数对结果（见附件数据 1），选取其中 500 个数据作为样本输入数据，建立神经网络模型。由于一个隐层的 BP 神经网络已经可以对任意的函数进行模拟，考虑建立只包含一层隐层的三层网络模型。又由于神经网络的输入数据是  $[-1, 1]$  区间，首先对数据进行归一化，常用的归一化方法是根据样本的取值范围来选择，问题中的学生和面试老师的数量的取值区间是所有整数，可能是趋向无穷大的取值，所以模型中用反正切函数将输入和输出数据归一为  $(0, 1)$  区间中，最小整数 1 变为 0.5，即为网络模型输入的最小值，

取值区间为 (0.5, 1)。

(1) 前 500 个数据作为样本训练模型

```
x=[1:500];  
y=[7 9 10 10 12 13 13 14 14 14 16 17 17 17 18.....];% (数据输入省略,  
具体数据见附件表格)
```

```
X=atan(x)*2/pi;  
Y=atan(y)*2/pi;  
threshold=[0.5 1];  
net=newff(threshold,[15,1],{'tansig','tansig','traigdx'});  
net.trainFcn='trainlm';  
net.trainParam.epochs=1000;  
net.trainParam.goal=0.0000001;  
net=train(net,X,Y);  
yy=sim(net,X)  
yTl=tan(1/2*pi*yy);  
plot(x,yTl, x, x, '*', x, y, '.');
```

样本的输入输出的数据个数都为 1，所以网络模型的输入层和输出层的节点数目都为 1，取隐层节点的数目为 15，采用双曲正切 S 型传递函数 tansig, 优化算法为梯度搜索 traigdx, 目标精度为 0.0000001, 最大迭代次数为 1000, 样本训练模型, 用得到的模型对样本数据的拟合结果见图 1。

TRAINLM, Epoch 0/1000, MSE 2.859e-007/1e-007, Gradient 0.00655493/1e-010

TRAINLM, Epoch 17/1000, MSE 9.13425e-008/1e-007, Gradient 0.0017508/1e-010

TRAINLM, Performance goal met.

模型对学生数为 379 时, 需要的老师数量为 78, 因此问题 2 中的 24 个老师的分配需要有一定的分配方案。

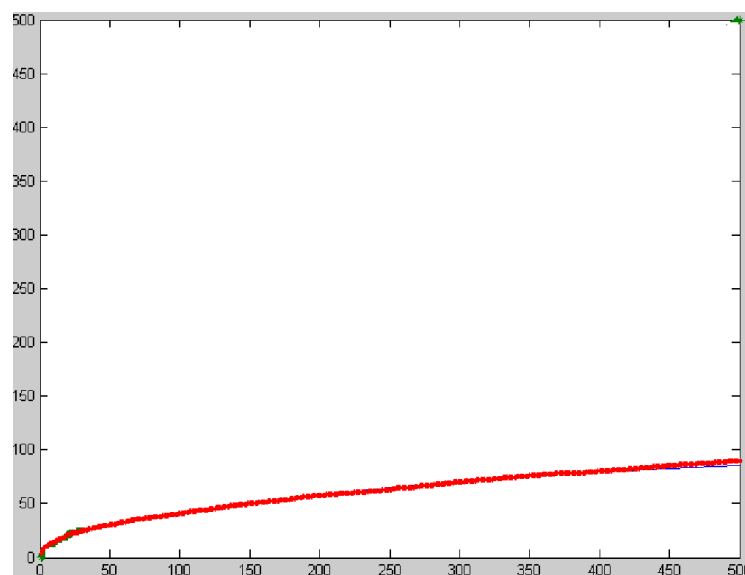


图 1 训练样本曲线与 BP 网络模型拟合结果

(2) 用前 500 个数据作为样本训练模型易见, 模型的精确度非常高, 用所得的模型对问题中的 501 到 700 个学生之间的数据对其所需要的老师数量进行

预测，所得结果见图 2。

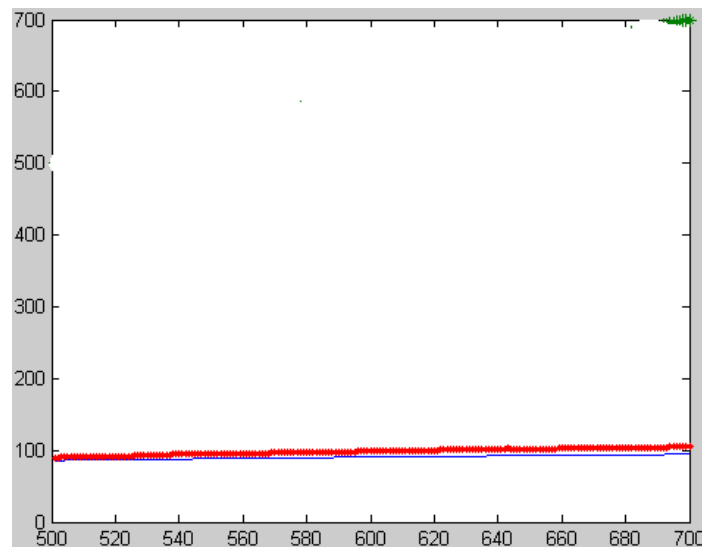


图 2 测试样本曲线与 BP 网络模型检验结果

用 501—700 个学生的数据作为模型检验样本

```
xt=[501:700];
```

```
yt=[90 90 91 91 91 91 91 91 91 91.....]; %(数据输入省略，具体数据见附件表格)
```

```
XT=atan(xt)*2/pi;
```

```
YT=atan(yt)*2/pi;
```

```
yyt=sim(net, XT);
```

```
yT2=tan(1/2*pi*yyt);
```

```
plot(xt, yyT2, xt, xt'*', xt, yt, '.');
```

检验结果，见图 2

- (3) 用所得到的总体 700 个数据（训练数据和测试数据）的散点图，分析数据分布的情况和发现规律

```
xz=[1:700];
```

```
yz=[y, yt];
```

```
XZ=[X, XT];
```

```
YZ=[Y, YT];
```

```
yyz=sim(net, XZ);
```

```
yyZ1=tan(1/2*pi*yyz);
```

```
plot(xz, yyZ1, xz, xz'*', xz, yz, '.');
```

结果见图 3（绿型为神经网络模型的数据散点图型，点型为模型二数据的散点图），数据的分布近似为二次曲线，因而可建立问题的解析模型，以求解问题。

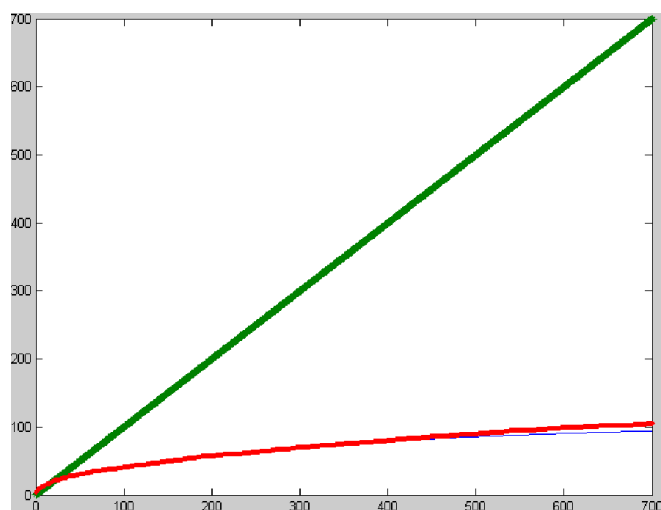


图 3 总样本曲线与 BP 网络模型结果

#### (4) 解析模型

设问题 1 学生数  $x$ ，所需要的最少老师数量为  $y$ ，从图中的曲线可以看出模型函数曲线的形状，曲线是具有二次曲线  $y = \sqrt{ax+b}$  ( $a, b$  为系数) 的相似形状，具有区间跳跃。随着  $x$  的变大，相邻的两个跳跃点之间是水平线段且长度变大，由于值域  $y$  的取值为整数，所以跳跃可能是对二次函数值取整引起的，因而可假设模型的函数形式为  $y$  为  $\sqrt{ax+b}$  的取整，这个假设具有合理性，由 3 可知建立解析模型  $y = \sqrt{ax+b}$ ，由总体 700 个数据求得相应的  $y^2$ ，系数  $a$  可看作变量  $x$ ， $y^2$  拟合曲线的斜率，由数据可知  $y^2$  与  $x$  的比值  $y^2/x$  近似为 18，假设模型曲线为  $y = \sqrt{18x+1}$ 。

在同一个图上画出三条曲线：样本曲线、BP 神经网络的模型曲线、 $y = \sqrt{18x+1}$  的二次曲线。

```
yyZ2=fix((18.*xz+1).^(1/2));
plot(xz,yyZ1, xz,xz,'*', xz,yyZ2,'-', xz,yz,'.');
```

结果见图 4，三条曲线的非常稳合（绿色型为神经网络模型的数据散点图型，点型为模型二次数据的散点图，线型为  $y = \sqrt{18x+1}$  的曲线散点图形），验证了解析模型  $y = \sqrt{18x+1}$  的合理性，也同时验证了三个模型对问题求解的有效型。

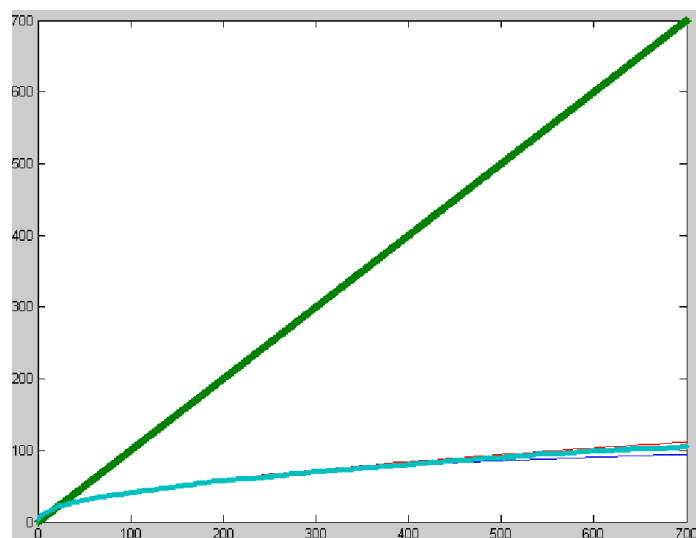


图 4 样本曲线、BP 神经网络的模型曲线、 $y = \sqrt{18x+1}$  的二次曲线模型结果

## ■ 问题 1.2

### 1) 0/1 整数规划方式求解

采用 0/1 整数规划求解该模型的算法见附件中算法 1。

其算法的优缺点同问题 1.1。

### 2) 组合数方式求解

同问题 1.1 的思路，考虑到问题要求求得老师数  $M$  至少为多大，即需要确定一个  $M$  的下限取值。为此用组合数方法分析满足问题条件的解的性质：

性质 1：可行解  $ST_{N \times M}$  每一行 4 个元素为 1 的（列）数位可以组成  $C_4^3$  个三元数位组；各行生成的  $C_4^3$  个（列）三元数位组组成数量为  $NC_4^3$  的数位组集合，在此集合中任何两个数位组都不相同。

例：假定第  $i$  行 4 个元素 1 分别位于第  $a, b, c, d$  列，则第  $i$  行的 4 个元素为 1 的数位可以组成  $(a, b, c), (a, b, d), (a, c, d), (b, c, d)$  4 个数位组。

证明：性质 1 中前一部分显而易见。后一部分的证明可以采用反证法：

假定所有的数对中存在 2 个数位组相同，则由于同一行的 4 个数中不可能产生相同的两个数位组，故这两个数位组必然产生于不同的行  $i$  和  $j$ 。由此可以推断第  $i$  行和第  $j$  行存在三个列数位相同的元素 1，即两个学生有三位面试老师相同，和题目约束条件矛盾。因此性质得证。

性质 2：可行解  $ST_{N \times M}$  满足  $NC_4^3 \leq C_M^3$

证明：由性质 1 可得，按照性质 1 的列数位对生成规则，所有行可以生成  $NC_4^3$  个两两不同的数位组。由于所有的数位组都是基于 1 到  $M$  的列号产生的三元组合  $(i, j, k) (i=1, \dots, m; j=1, \dots, m; k=1, \dots, m; i \neq j, j \neq k \text{ 且 } i \neq k)$ ，故其数目不会超过 1 到  $M$  能够产生的三元组合最大数目  $C_M^3$ 。故性质 2 得证。

由性质 2 可以推出  $M$  的取值下界条件满足：

$$M^3 - 3M^2 + 2M - 24N \geq 0 \quad (6)$$

即结论： $M$  至少应使  $M^3 - 3M^2 + 2M - 24N \geq 0$  成立时，才可能满足任两位学生的“面试组”都没有三位面试老师相同。

由于上述计算过程只考虑了该问题的必要性条件，因此求出的下界不一定是问题的下确界，更为精确的下确界模型为  $NC_4^3 \leq C_M^3 - t$ ，其中  $t$  为关于  $N$  和  $M$  的函数。

### 3) 基于最短路径的阶段递进搜索算法

- ✓ 假定 1：存在  $N$  个学生对应  $M$  位老师的最优分配方案之一的情况下，该最优分配方案满足各老师的面试负荷量相对均衡；
- ✓ 假定 2：存在满足假定 1 的  $ST_{(N+1) \times M}$  的最优分配方案，该方案可由某一个满足假定 1 的  $ST_{N \times M}$  的最优分配方案在不改变现有分配的基础上扩展取得（即  $ST_{(N+1) \times M}$  中存在一个  $N \times M$  子矩阵等于  $ST_{N \times M}$ ）；

由假定 1 和假定 2 可以得出以下两条搜索规则：

规则 1：从  $ST_{N \times M}$  的某一最优分配方案向  $ST_{(N+1) \times M}$  的某一最优方案扩展时，只需在  $ST_{N \times M}$  的基础上增加第  $N+1$  行，并调整其中非零元素的位置（同时方案根据需要增加一定数量的列或者保持列数不变）；

规则 2：在对  $N+1$  个学生进行老师分配时，总是先选取在  $N$  个学生数下所有老师中面试任务最轻的第一位老师；

具体算法思想同问题 1.1 中的实现方式，算法实现略。

## ◆ 问题 2

### 1. 问题分析：

根据问题描述，此问题需要满足目标条件 Y1, Y2, Y3, Y4 的满意解，故属于一个多目标的决策模型，其解决思路可以应用多目标规划的实现方法。

常用的多目标规划实现方法有多种，实际中应用较多且较简便的是通过加权把多目标决策转化为单目标问题。综合考虑算法复杂性和性能，本问题将采用加权方法。

### 2. 模型求解

本例算法由于学生数目较多，采用传统的遍历式搜索比较的目标决策方法受限于设计和运行环境，其可行性受到较大影响。为了提高其算法可行性，本例中对于算法做出以下改进：

- 1) 把个目标条件的满意度转化为数量指标，通过一定的权重分配把多个目标的数量指标转化为单目标的数量指标。
- 2) 改变原有的遍历式搜索算法，选用从可行解空间中随机抽取可行解子集的方式，从可行解子集中寻找满意解；
- 3) 当可行解子集覆盖数量越多时，找到的满意解更趋近于整个问题的满意解；
- 4) 通过增加算法的有效迭代次数，可以增加抽取的可行解子集覆盖的可行解数量。

具体的算法实现详见附件算法 3。

由于运行环境的限制，我们选取了 105 次作为有效迭代次数，并在每迭代 5 次的基础上打印出个最优化分配方案，相应的优化情况以表格形式列举如下：

目标条件		Y1	Y3	Y4	总计目标	最优目标
调整加权		1	0.0001	0.5		17.7454
有效迭代次数	95	7.4054	47753	18	21.1807	17.7454
	96	6.4043	47594	19	20.6637	17.7454
	97	4.1772	47323	16	16.9095	16.9095
	98	6.3702	47589	19	20.6291	16.9095
	99	9.2345	48103	16	22.0448	16.9095
	100	6.5916	47622	17	19.8538	16.9095
	101	7.1485	47710	18	20.9195	16.9095
	102	7.0874	47700	17	20.3574	16.9095
	103	7.5046	47770	17	20.7816	16.9095
	104	9.7386	48213	19	24.0599	16.9095
	105	8.0145	47861	17	21.3006	16.9095

从迭代结果可以归纳出以下几点：

- 1) 由于采取随机矩阵生成并比较优化方式，在算法复杂度和计算环境的影响决定了迭代次数不可能太多，导致优化次数也相对较少。这一点可以通过矩阵的随机生成原理作出一定的解释。随即生成下，每个老师的面试任务负荷量应该相对平均，得出的解较趋向于满意解，使得在相对于样本空间较小的样本数条件下，命中的可以进一步优化的几率会较少；



- 2) 由于个目标约束数字表示的数量级不同, 为了使各个目标在总目标中都具有相对均衡的影响, 必须要通过加权调整, 是的加权后各目标数量相对接近;
- 3) 由于 Y2 是一个必要目标, 而非满意目标, 故此应该把 Y2 作为可行解的判别条件, 而非满意解的判别条件;

具体的分配方案见附件数据 2。

从方案排列以及各目标的数量表示和其他方案下各目标的数量表示比较来看, 可以得出以下几点:

- 1) 对于 Y1, Y3, Y4 三个目标约束而言, 最终方案相较于邻近方案在三个目标上面都由优化 (具体表现为三个目标的数量表示都比其它相邻方案小);
- 2) 由于加权的设定, 以及计算出的 Y1, Y3, Y4 数量表示的数量极, 可以看出 Y1, Y4 目标偏好度相当, Y3 相对而言目标偏好度较低, 此决定是由权重的设定导致, 并不是由最终方案的特定导致;
- 3) 由于上面 (2) 特性的存在, 可以认为最终方案在优化前提上就更多考虑了目标 Y1, Y4 的满意度;
- 4) Y2 目标由于是一个确定性 (可行性) 约束, 最终方案必然是完全满足了条件 Y2。

### ◆ 问题 3

#### 1. 问题分析:

假设面时老师中理科与文科的老师各占一半, 并且要求每位学生接受两位文科与两位理科老师的面试。在此条件下, 分别对问题一和问题二进行分析:

问题一: 等同于在问题一满足现有的约束条件基础上 (即任两位学生的“面试组”都没有两位 (子问题 1) 和三位 (子问题 2) 面试老师相同的情形), 增加一个新的约束条件, 对于问题一的子问题 1, 2, 新增加的约束分别为:

子问题 1: 可以把  $ST_{N \times M}$  矩阵划分为两个  $ST_{N \times \frac{M}{2}}$  矩阵, 两个  $ST_{N \times \frac{M}{2}}$  都满足每行上有且只有两个 1, 任何两行不完全相同; (M 为偶数, 下同)

子问题 2: 可以把  $ST_{N \times M}$  矩阵划分为两个  $ST_{N \times \frac{M}{2}}$  矩阵, 两个  $ST_{N \times \frac{M}{2}}$  都满足每行上有且只有两个 1;

问题二: 由于文科和理科的老师各占一半, 假定可以通过列交换把文科老师移动到  $1, \dots, M/2$  列, 理科老师移动到  $1+M/2, \dots, M$  列, 则有其满足条件: 可以把  $ST_{N \times M}$  矩阵划分为两个  $ST_{N \times \frac{M}{2}}$  矩阵, 两个  $ST_{N \times \frac{M}{2}}$  都满足每行上有且只有两个 1。

#### 2. 模型求解

##### 1) 问题一:

##### (1) 子问题 1:

类似于问题一中求解的子问题 1.1, 考虑 M 取值的下界条件。

性质 A:  $N \leq C_{M/2}^2$

证明: 仿照问题一的子问题 1.1 中性质 2 证明,  $ST_{N \times \frac{M}{2}}$  满足每行上有且只有

两个 1，任何两行不完全相同，则 N 行上共有  $N$  个每行上非零元素列号生成的二元数对，且这些二元数对两两不等。由于  $M/2$  个列号位置能生成的二元数对最多有  $C_{M/2}^2$  个，故性质 A 得证。

则该问题需要同时满足性质 A 和问题一中子问题 1.1 的性质 2，故其必要条件为：

$$\begin{cases} N \leq C_{M/2}^2 \\ NC_4^2 \leq C_M^2 \end{cases}$$

由于  $M \leq 4 * N$ ，

$$\text{故有 } M \geq \frac{1 + \sqrt{1 + 48N}}{2}$$

(2) 子问题 2:

2) 问题二:

根据前面的可以把  $ST_{N \times M}$  矩阵划分为两个  $ST_{N \times \frac{M}{2}}$  矩阵，两个  $ST_{N \times \frac{M}{2}}$  都满足每行上有且只有两个 1。参照前述附件算法 3，如果仍然采取随机抽样筛选局部满意解的方法，则只需要使随机出来的可行解满足前  $M/2$  列组成的子矩阵每行上只有两个非零元素。

为了操作方便，可以仅把附件算法 3 的随机矩阵生成规则修改如下：

- (1) 随机生成元素为  $ST_{N \times \frac{M}{2}}$  矩阵，满足条件各行元素相加为 2 则保留；
- (2) 随机生成两个有效的  $ST_{N \times \frac{M}{2}}$  矩阵，组合成一个  $ST_{N \times M}$ ，判断该矩阵是否满足条件 Y2，如果满足，则继续，否则考察下一个随机矩阵；
- (3) 满足条件 Y2 的基础上，进一步判定该矩阵对于 Y1, Y3, Y4 的满意程度，并和之前取得的满意解进行比较。

#### ◆ 问题 4

所谓考生与面试老师之间分配的均匀性可以从两方面理解，一是每个考生被相同数目的老师面试；二是每位老师面试的学生数目相等。现在根据附件算法 2（基于最短路径的阶段递进搜索算法）的运行结果可以发现：

当学生数 N 确定且任两位学生的“面试组”都没有两位老师相同的情形下，若 M 为学生数为 N 时对应的最少的老师数，此时在这 M 个老师中，任意两位老师面试的学生数差额不超过 3。一旦遇到再增加 1 个学生必须至少增加 1 个老师的情况时，新增老师的面试学生数与其他 M 个老师的面试学生数有显著差异，而这种显著差异随着 N 的不断增大而逐渐缩小。

所谓面试公平性是许多学生都被同一位老师面试，这是由于该老师自身对不同学生的主观印象或意见而产生不公平的评判。若在老师面试学生数目不变的情况下，为了提高面试的公平性，必须增加老师的数目，使得每个学生尽可能被更多的老师面试。可见，随着老师数目的增多，考生与老师之间的分配正逐步向均匀性逼近。

综上所述，在任两位学生的“面试组”都没有两位老师相同，且 M 为学生数为 N 时对应的最少的老师数的情形下，考生与面试老师之间分配的均匀性与面试公平性存在正相关关系。除此之外，当 N 增加 1，考生与面试老师之间分配的均匀性与面试公平性之间的平衡状态被破坏，并随着 N 的逐渐增大，这种平衡状态

又逐渐恢复；反之，当  $N$  减小 1，考生与面试老师之间分配的均匀性与面试公平性之间的平衡状态也被破坏，并随着  $N$  的逐渐减小，这种平衡状态逐渐恶化，直至当  $N$  减小到面试分配矩阵中至少有 1 位老师的面试学生只有 1 时，在此基础上进一步减少一个学生后的面试分配矩阵又达到了新的平衡状态。

可见，要保证面试的公平性可以充分利用师资力量，但同时这也存在着两个矛盾，一是当学生数目不断增大时，每个老师的面试学生数也在不断增大，这对老师的面试质量提出了疑问；其次，随着学生数目的增大，要保证面试的公平性需要增加一定数量的老师，这必将增加面试的成本，且对面试工作的组织管理工作提出了巨大挑战。

因此，在成本、控制能力、时间的多重约束下，可以重新考虑如何在必需的面试老师数目的条件下合理分配老师的专业背景，从而提高老师面试的质量及公平性，避免了盲目分配给学生专业不对口的老师。此外，还可以根据学生的学习情况，把学生分成若干等级，把相关老师分配给给定等级的学生群进行面试；还可以要求每个面试组内必须加一个权威老师等等。

#### 四、 附件：

##### 1 附件算法 1

```
function [x,fval]=linprog01(N)
% [x,fval]=linprog01(N)函数求得学生数为 N 时所需的最少老师的解，为问题 1 的求解程序
% x, fval 为最优解的返回值
% 其中 x 为学生与老师的匹配向量，fval 为最少老师的个数
tic
init_count=3*N+1;% 初始化所需要的老师最大数量
var_count=init_count*N;% 配对方案的变量个数
deploy_instance=zeros(var_count,1);
x=deploy_instance;
opt_solution=init_count;
p=2^(init_count*(N-1)+3)+2^(init_count*(N-1)+2)+2^(init_count*(N-1)+1)+2^(init_count*(N-1)
);
% 利用每位学生都对应 4 个老师，压缩搜索空间
for i=p:2^var_count-1 % 利用 2 进制变换遍历所有配对方案空间
    % strBin_i=dec2bin(i);
    % deploy_instance=zeros(var_count,1);
    %for k=1:length(strBin_i)
    %    deploy_instance(k)=str2num(strBin_i(k));
    % end
    deploy_instance=de2bi(i,var_count)'; % 可行的方案必需满足变量之和为 4*N
    if ~(sum(deploy_instance)==4*N)
        continue
    end
    stud_teacher=zeros(N,init_count);
    for m=1:N % 向量转化为配对方案矩阵
        stud_teacher(m,(1:init_count))=deploy_instance(((m-1)*init_count+1):m*init_count);
    end
    constrA=(sum(stud_teacher')==4); %每个学生配对 4 个老师

    if ~all(constrA)
        continue;
    end

    sum_tow_lines=zeros(N*(N-1)/2,1);
    s=0;
    for m=1:N-1 % 统计任意两个学生对应的相同老师的情况
        for n=m+1:N
            s=s+1;
            sum_temp=stud_teacher(m,:)+stud_teacher(n,:);
            sum_temp=(sum_temp>1);
            sum_tow_lines(s,1)=sum(sum_temp);
        end
    end
end
```

```

        end
    end
    constrB=(sum_tow_lines<2);% 满足任两位学生没有两位面试老师相同
    if all(constrB)
        sum_column_result=(sum(stud_teacher)>0);% 统计需要的老师数量
        objVal=sum(sum_column_result);
        if objVal<=opt_solution
            opt_solution=objVal;
            x=deploy_instance;
        end
    end
end
end
toc
fval=opt_solution;

```

## 2 附件算法 2

```

package test;
public class Test {
    private static int x[][] = new int[1000][1000]; // x[i][j]表示第i个
    学生与第j位老师的关系，若x[i][j]=1，表示被面试，x[i][j]=0表示没有
    private static final int MAX_TEA_FOR_STUD = 4;
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        int n = 0;
        try {
            n = Integer.parseInt(args[0]);
        } catch (NumberFormatException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        for (int i = 501; i <= 700; i++)
            test(i);
    }

    private static int test(int N) {
        int m; // m表示动态增加教师时用到的总列数
        int n = 0; // n表示动态增加学生时用到的总列数
        int t_n[] = new int[10000]; // T_n[j]表示第j位老师面试的学生数
        int g; // g表示增加一个学生，需要分配给他的老师的号码
        n = 1; // 初始值
        m = 4; // 初始值
        int s_m = 0;
    }
}

```

号码

```
for (int a = 0; a < N; a++)
    for (int b = 0; b < 100; b++)
        x[a][b] = 0;
x[0][0] = x[0][1] = x[0][2] = x[0][3] = 1;
while (n < N) {
    for (int b = 0; b < m; b++)
        // 计算原来每位老师面试的学生数
        t_n[b] = 0;
    for (int b = 0; b < m; b++)
        // 计算原来每位老师面试的学生数
        for (int c = 0; c < n; c++)
            if (x[c][b] == 1)
                t_n[b]++;
    int t_n_temp;
    int m_kong = m;
    while (s_m < MAX_TEA_FOR_STUD) {
        g = 0;
        t_n_temp = N;
        for (int b = 0; b < m; b++) // 计算分配给新增学生的第一位教师
        {
            if ((x[n][b] == 0) && (t_n[b] < t_n_temp)) {
                g = b;
                t_n_temp = t_n[b];
            }
        }
        m_kong = add_delete(n, m, g, m_kong);
        s_m++;
        if (m_kong == 0)
            break;
    }
    for (int i = 0; i < m; i++) {
        if (x[n][i] == 2)
            x[n][i] = 0;
    }
    if (s_m < 4) {
        for (int i = 0; i < 4 - s_m; i++) {
            x[n][m++] = 1;
        }
    }
    n++;
    s_m = 0;
}
System.out.println(m);
```

```

        return m;
    }

    private static int add_delete(int n, int m, int b, int init_m_kong)
    {
        x[n][b] = 1; // 给新增的学生指派第一位教师
        // int s_m = 1; // m个老师
        int m_kong = init_m_kong - 1;
        for (int e = 0; e < n; e++) // 找出不可添加给学生n的老师号码，并将此
        老师标志为2
        {
            if (x[e][b] == 1)
                for (int f = 0; f < m; f++)
                    if (x[e][f] == 1 && f != b && x[n][f] == 0) {
                        x[n][f] = 2;
                        m_kong--;
                    }
        }
        return m_kong;
    }
}

```

### 3 附件算法 3

```

function [t1,t3,t4,stud_teacher,fval]=linprog02(N,Max_times)
% function [t1,t3,t4,stud_teacher,fval]=linprog02(N,Max_times)为问题 2 的求解程序
% 输入 N,Max_times 分别为学生个数和程序进行搜索迭代的次数
% 输出 stud_teacher,fval 分别为分配方案的配对矩阵和对应的总目标函数值
% 输出 t1,t3,t4 分别为对应的条件 1，条件 3，条件 4 的目标函数值
tic
init_count=24;%初始的老师数量
var_count=init_count*N;
x=zeros(N,init_count);% 构造学生老师的配对矩阵
opt_solution=inf;
t1=0;
t3=0;
t4=0;
q1=1;
q2=0.0001;
q3=0.5;
times_count=0;
print_times=1;
while 1 %伪随机生成随机方案（对每个学生配对 4 个老师）
    stud_teacher=zeros(N,init_count);

```

```

for(i=1:N) % 随机生成两个点的位置
    %while 1
        %instance=(rand(1,init_count)>0.82);
        %if(sum(instance)==4)
            % stud_teacher(i,(1:init_count))=instance;
            % break;
        %end
    %end
    a=ceil(rand()*init_count);
    b=ceil(rand()*init_count);
    while(b==a)
        b=ceil(rand()*init_count);
    end
    c=ceil(rand()*init_count); %随机生成第三个点的位置
    while((c==a)||(c==b))
        c=ceil(rand()*init_count);
    end
    d=ceil(rand()*init_count); % 随机生成第四个点的位置
    while((d==a)||(d==b)||(d==c))
        d=ceil(rand()*init_count);
    end
    stud_teacher(i,a)=1;
    stud_teacher(i,b)=1;
    stud_teacher(i,c)=1;
    stud_teacher(i,d)=1;
end
sum_tow_lines=zeros(N*(N-1)/2,1);
M=init_count;
sum_tow_cols=zeros(M*(M-1)/2,1);
s=0;
for m=1:M-1 % 统计任意两位考官面试相同学生的数量
    for n=m+1:M
        s=s+1;
        sum_temp=stud_teacher(:,m)+stud_teacher(:,n);
        sum_temp=(sum_temp>1);
        sum_tow_cols(s,1)=sum(sum_temp);
    end
end
t4=max(sum_tow_cols); % 最大两位考官面试相同学生的数量的值
s=0;
flag2=0;
for m=1:N-1 % 统计任两位考生中有相同考官的情况
    for n=m+1:N
        s=s+1;

```



```

        sum_temp=stud_teacher(m,:)+stud_teacher(n,:);
        sum_temp=(sum_temp>1);
        sum_tow_lines(s,1)=sum(sum_temp);
        if ~(sum_tow_lines(s,1)<4) % 不同考生的考官不能完全相同
            flag2=1;
            break;
        end
    end
    if flag2==1
        break;
    end
end
if flag2==1
    continue;
end
t3=sum(sum_tow_lines);% 考生中有相同考官的情形总数量
t1=std((sum(stud_teacher))); % 每位考官面试的学生数量的标准差
objVal=t1*q1+t3*q2+t4*q3; % 总目标值
t1
t3
t4
if objVal<=opt_solution % 优的方案的迭代
    opt_solution=objVal;
    x=stud_teacher;
end
times_count=times_count+1
if(times_count>=print_times*5) % 每迭代 5 次，打印方案结果
    opt_solution
    x
    print_times=print_times+1;
end
if(times_count>Max_times)
    break;
end
end
end
toc
fval=opt_solution;
end

```

#### 4 附件数据1

N=1	M=4	N=2	M=7	N=3	M=9	N=4	M=10	N=5	M=10
N=6	M=12	N=7	M=13	N=8	M=13	N=9	M=14	N=10	M=14
N=11	M=14	N=12	M=16	N=13	M=17	N=14	M=17	N=15	M=17

N=16	M=18	N=17	M=18	N=18	M=19	N=19	M=20	N=20	M=21
N=21	M=21	N=22	M=22	N=23	M=22	N=24	M=22	N=25	M=23
N=26	M=23	N=27	M=23	N=28	M=23	N=29	M=24	N=30	M=24
N=31	M=25	N=32	M=25	N=33	M=25	N=34	M=25	N=35	M=26
N=36	M=27	N=37	M=27	N=38	M=28	N=39	M=28	N=40	M=28
N=41	M=29	N=42	M=29	N=43	M=29	N=44	M=29	N=45	M=30
N=46	M=30	N=47	M=30	N=48	M=30	N=49	M=30	N=50	M=31
N=51	M=31	N=52	M=31	N=53	M=31	N=54	M=31	N=55	M=32
N=56	M=32	N=57	M=33	N=58	M=33	N=59	M=33	N=60	M=33
N=61	M=33	N=62	M=34	N=63	M=34	N=64	M=34	N=65	M=35
N=66	M=35	N=67	M=35	N=68	M=35	N=69	M=36	N=70	M=36
N=71	M=36	N=72	M=36	N=73	M=36	N=74	M=36	N=75	M=37
N=76	M=37	N=77	M=37	N=78	M=37	N=79	M=37	N=80	M=37
N=81	M=38	N=82	M=38	N=83	M=38	N=84	M=38	N=85	M=38
N=86	M=38	N=87	M=38	N=88	M=39	N=89	M=39	N=90	M=39
N=91	M=39	N=92	M=39	N=93	M=40	N=94	M=40	N=95	M=40
N=96	M=40	N=97	M=40	N=98	M=40	N=99	M=41	N=100	M=41
N=101	M=41	N=102	M=41	N=103	M=41	N=104	M=42	N=105	M=42
N=106	M=42	N=107	M=43	N=108	M=43	N=109	M=43	N=110	M=43
N=111	M=43	N=112	M=43	N=113	M=44	N=114	M=44	N=115	M=44
N=116	M=44	N=117	M=44	N=118	M=44	N=119	M=44	N=120	M=44
N=121	M=45	N=122	M=45	N=123	M=45	N=124	M=45	N=125	M=45
N=126	M=46	N=127	M=46	N=128	M=46	N=129	M=46	N=130	M=46
N=131	M=47	N=132	M=47	N=133	M=47	N=134	M=47	N=135	M=47
N=136	M=48	N=137	M=48	N=138	M=48	N=139	M=48	N=140	M=48
N=141	M=48	N=142	M=49	N=143	M=49	N=144	M=49	N=145	M=49
N=146	M=49	N=147	M=49	N=148	M=49	N=149	M=50	N=150	M=50
N=151	M=50	N=152	M=50	N=153	M=50	N=154	M=50	N=155	M=50
N=156	M=50	N=157	M=51	N=158	M=51	N=159	M=51	N=160	M=51
N=161	M=51	N=162	M=51	N=163	M=52	N=164	M=52	N=165	M=52
N=166	M=52	N=167	M=52	N=168	M=53	N=169	M=53	N=170	M=53
N=171	M=53	N=172	M=54	N=173	M=54	N=174	M=54	N=175	M=54
N=176	M=54	N=177	M=54	N=178	M=54	N=179	M=55	N=180	M=55
N=181	M=55	N=182	M=55	N=183	M=56	N=184	M=56	N=185	M=56
N=186	M=56	N=187	M=56	N=188	M=57	N=189	M=57	N=190	M=57
N=191	M=57	N=192	M=57	N=193	M=57	N=194	M=57	N=195	M=58
N=196	M=58	N=197	M=58	N=198	M=58	N=199	M=58	N=200	M=58
N=201	M=58	N=202	M=58	N=203	M=58	N=204	M=58	N=205	M=59
N=206	M=59	N=207	M=59	N=208	M=59	N=209	M=59	N=210	M=59
N=211	M=59	N=212	M=59	N=213	M=59	N=214	M=60	N=215	M=60
N=216	M=60	N=217	M=60	N=218	M=60	N=219	M=60	N=220	M=60
N=221	M=60	N=222	M=60	N=223	M=60	N=224	M=60	N=225	M=61
N=226	M=61	N=227	M=61	N=228	M=61	N=229	M=61	N=230	M=61
N=231	M=61	N=232	M=61	N=233	M=61	N=234	M=61	N=235	M=62

N=236	M=62	N=237	M=62	N=238	M=62	N=239	M=62	N=240	M=62
N=241	M=62	N=242	M=62	N=243	M=62	N=244	M=62	N=245	M=63
N=246	M=63	N=247	M=63	N=248	M=63	N=249	M=63	N=250	M=63
N=251	M=63	N=252	M=64	N=253	M=64	N=254	M=64	N=255	M=65
N=256	M=65	N=257	M=65	N=258	M=65	N=259	M=65	N=260	M=65
N=261	M=65	N=262	M=65	N=263	M=65	N=264	M=65	N=265	M=65
N=266	M=65	N=267	M=65	N=268	M=66	N=269	M=66	N=270	M=66
N=271	M=66	N=272	M=67	N=273	M=67	N=274	M=67	N=275	M=67
N=276	M=67	N=277	M=67	N=278	M=67	N=279	M=67	N=280	M=67
N=281	M=67	N=282	M=68	N=283	M=68	N=284	M=68	N=285	M=68
N=286	M=68	N=287	M=68	N=288	M=68	N=289	M=68	N=290	M=69
N=291	M=69	N=292	M=69	N=293	M=69	N=294	M=69	N=295	M=69
N=296	M=70	N=297	M=70	N=298	M=70	N=299	M=70	N=300	M=70
N=301	M=70	N=302	M=70	N=303	M=70	N=304	M=71	N=305	M=71
N=306	M=71	N=307	M=71	N=308	M=71	N=309	M=71	N=310	M=71
N=311	M=71	N=312	M=72	N=313	M=72	N=314	M=72	N=315	M=72
N=316	M=72	N=317	M=72	N=318	M=72	N=319	M=72	N=320	M=72
N=321	M=72	N=322	M=72	N=323	M=72	N=324	M=73	N=325	M=73
N=326	M=73	N=327	M=73	N=328	M=73	N=329	M=73	N=330	M=73
N=331	M=74	N=332	M=74	N=333	M=74	N=334	M=74	N=335	M=74
N=336	M=74	N=337	M=74	N=338	M=74	N=339	M=74	N=340	M=74
N=341	M=75	N=342	M=75	N=343	M=75	N=344	M=75	N=345	M=75
N=346	M=75	N=347	M=75	N=348	M=75	N=349	M=75	N=350	M=76
N=351	M=76	N=352	M=76	N=353	M=76	N=354	M=76	N=355	M=76
N=356	M=76	N=357	M=76	N=358	M=76	N=359	M=77	N=360	M=77
N=361	M=77	N=362	M=77	N=363	M=77	N=364	M=77	N=365	M=77
N=366	M=77	N=367	M=77	N=368	M=78	N=369	M=78	N=370	M=78
N=371	M=78	N=372	M=78	N=373	M=78	N=374	M=78	N=375	M=78
N=376	M=78	N=377	M=78	N=378	M=78	N=379	M=78	N=380	M=78
N=381	M=78	N=382	M=78	N=383	M=78	N=384	M=79	N=385	M=79
N=386	M=79	N=387	M=79	N=388	M=79	N=389	M=79	N=390	M=79
N=391	M=80	N=392	M=80	N=393	M=80	N=394	M=80	N=395	M=80
N=396	M=80	N=397	M=80	N=398	M=80	N=399	M=81	N=400	M=81
N=401	M=81	N=402	M=81	N=403	M=81	N=404	M=81	N=405	M=81
N=406	M=81	N=407	M=81	N=408	M=81	N=409	M=82	N=410	M=82
N=411	M=82	N=412	M=82	N=413	M=82	N=414	M=82	N=415	M=82
N=416	M=82	N=417	M=82	N=418	M=82	N=419	M=82	N=420	M=82
N=421	M=83	N=422	M=83	N=423	M=83	N=424	M=83	N=425	M=83
N=426	M=83	N=427	M=83	N=428	M=83	N=429	M=84	N=430	M=84
N=431	M=84	N=432	M=84	N=433	M=84	N=434	M=84	N=435	M=84
N=436	M=84	N=437	M=84	N=438	M=85	N=439	M=85	N=440	M=85
N=441	M=85	N=442	M=85	N=443	M=85	N=444	M=85	N=445	M=85
N=446	M=85	N=447	M=85	N=448	M=86	N=449	M=86	N=450	M=86
N=451	M=86	N=452	M=86	N=453	M=86	N=454	M=86	N=455	M=86

N=456	M=86	N=457	M=87	N=458	M=87	N=459	M=87	N=460	M=87
N=461	M=87	N=462	M=87	N=463	M=87	N=464	M=87	N=465	M=87
N=466	M=87	N=467	M=87	N=468	M=87	N=469	M=87	N=470	M=88
N=471	M=88	N=472	M=88	N=473	M=88	N=474	M=88	N=475	M=88
N=476	M=88	N=477	M=88	N=478	M=88	N=479	M=88	N=480	M=88
N=481	M=88	N=482	M=89	N=483	M=89	N=484	M=89	N=485	M=89
N=486	M=89	N=487	M=89	N=488	M=89	N=489	M=89	N=490	M=89
N=491	M=89	N=492	M=90	N=493	M=90	N=494	M=90	N=495	M=90
N=496	M=90	N=497	M=90	N=498	M=90	N=499	M=90	N=500	M=90
N=501	M=90	N=502	M=90	N=503	M=91	N=504	M=91	N=505	M=91
N=506	M=91	N=507	M=91	N=508	M=91	N=509	M=91	N=510	M=91
N=511	M=91	N=512	M=91	N=513	M=91	N=514	M=91	N=515	M=91
N=516	M=92	N=517	M=92	N=518	M=92	N=519	M=92	N=520	M=92
N=521	M=92	N=522	M=92	N=523	M=92	N=524	M=92	N=525	M=92
N=526	M=93	N=527	M=93	N=528	M=93	N=529	M=93	N=530	M=93
N=531	M=93	N=532	M=94	N=533	M=94	N=534	M=94	N=535	M=94
N=536	M=94	N=537	M=94	N=538	M=95	N=539	M=95	N=540	M=95
N=541	M=95	N=542	M=95	N=543	M=95	N=544	M=95	N=545	M=95
N=546	M=95	N=547	M=95	N=548	M=95	N=549	M=95	N=550	M=95
N=551	M=95	N=552	M=96	N=553	M=96	N=554	M=96	N=555	M=96
N=556	M=96	N=557	M=96	N=558	M=96	N=559	M=96	N=560	M=96
N=561	M=96	N=562	M=96	N=563	M=96	N=564	M=96	N=565	M=96
N=566	M=96	N=567	M=96	N=568	M=96	N=569	M=97	N=570	M=97
N=571	M=97	N=572	M=97	N=573	M=97	N=574	M=97	N=575	M=97
N=576	M=97	N=577	M=97	N=578	M=97	N=579	M=97	N=580	M=97
N=581	M=97	N=582	M=98	N=583	M=98	N=584	M=98	N=585	M=98
N=586	M=98	N=587	M=98	N=588	M=98	N=589	M=98	N=590	M=98
N=591	M=98	N=592	M=98	N=593	M=98	N=594	M=98	N=595	M=98
N=596	M=99	N=597	M=99	N=598	M=99	N=599	M=99	N=600	M=99
N=601	M=99	N=602	M=99	N=603	M=99	N=604	M=99	N=605	M=99
N=606	M=99	N=607	M=99	N=608	M=99	N=609	M=99	N=610	M=99
N=611	M=100	N=612	M=100	N=613	M=100	N=614	M=100	N=615	M=100
N=616	M=100	N=617	M=100	N=618	M=100	N=619	M=100	N=620	M=100
N=621	M=100	N=622	M=101	N=623	M=101	N=624	M=101	N=625	M=101
N=626	M=101	N=627	M=101	N=628	M=101	N=629	M=101	N=630	M=101
N=631	M=101	N=632	M=101	N=633	M=101	N=634	M=101	N=635	M=101
N=636	M=101	N=637	M=101	N=638	M=101	N=639	M=101	N=640	M=102
N=641	M=102	N=642	M=102	N=643	M=103	N=644	M=102	N=645	

M=102  
N=646 M=102 N=647 M=102 N=648 M=102 N=649 M=102 N=650  
M=102  
N=651 M=102 N=652 M=102 N=653 M=102 N=654 M=102 N=655  
M=102  
N=656 M=102 N=657 M=102 N=658 M=102 N=659 M=103 N=660  
M=103  
N=661 M=103 N=662 M=103 N=663 M=103 N=664 M=103 N=665  
M=103  
N=666 M=103 N=667 M=103 N=668 M=103 N=669 M=103 N=670  
M=103  
N=671 M=103 N=672 M=103 N=673 M=103 N=674 M=103 N=675  
M=103  
N=676 M=103 N=677 M=103 N=678 M=103 N=679 M=104 N=680  
M=104  
N=681 M=104 N=682 M=104 N=683 M=104 N=684 M=104 N=685  
M=104  
N=686 M=104 N=687 M=104 N=688 M=104 N=689 M=104 N=690  
M=104  
N=691 M=104 N=692 M=104 N=693 M=104 N=694 M=105 N=695  
M=105  
N=696 M=105 N=697 M=105 N=698 M=105 N=699 M=105 N=700  
M=105

## 5 附件数据 2

满意解目标函数值

opt\_solution =

16.9095

满意解分配方案

x =

Columns 1 through 14

0	0	0	0	1	0	1	0	0	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	1	0
1	0	0	0	0	0	1	1	0	0	0	0	0	1
0	0	1	0	0	0	0	1	1	0	0	0	0	1

0	1	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	1	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	1
1	0	0	0	0	0	1	0	1	0	0	0	0	0
0	0	0	0	0	0	1	1	0	0	1	1	0	0
0	0	1	0	0	1	0	0	0	1	0	0	0	0
1	0	0	1	0	0	0	0	1	1	0	0	0	0
1	0	1	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	1	0	0	0	0	0	1
0	0	0	1	0	1	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	1	1	1	0	1	0
0	0	0	0	0	0	0	0	1	1	0	0	1	0
0	1	1	0	0	0	0	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	1	0	1	0	0	0	1	0	0	0
0	0	0	1	0	1	0	0	1	0	0	0	0	0
0	0	1	0	0	1	0	0	0	0	0	1	0	0

0	1	0	0	1	0	0	1	0	0	0	0	0	0
1	0	0	1	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1	0	0	0	1	1
0	0	0	0	0	1	0	1	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	1	1	0
1	1	0	0	0	0	1	0	0	1	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	1	0	1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	1	0	0	0	0	0	0	1
0	0	0	0	0	1	0	1	0	0	0	0	1	0
1	1	0	0	0	0	0	0	1	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0	0	1	0	0
1	0	1	0	0	0	0	0	0	0	1	1	0	0
1	0	0	0	0	1	0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	0	0	0	0
0	0	0	0	0	0	0	1	0	1	0	0	0	0
0	1	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	1	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	1
0	0	0	0	1	0	0	0	0	0	0	0	0	0

1	0	0	0	1	0	0	0	0	1	0	0	0	0
1	0	0	0	0	0	1	0	0	0	0	0	0	0
1	0	0	0	0	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	1	0	0	0	0	0	1
0	1	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	1	0	0	0	1	0	0	0	0	0	0
0	1	0	0	0	0	1	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	1	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0
1	0	0	1	1	0	0	0	0	0	1	0	0	0
0	0	0	0	0	1	0	0	0	0	1	1	0	0
0	0	0	0	0	0	1	1	1	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	1	0	0
0	0	0	0	1	0	0	0	0	1	0	0	0	0
0	0	0	0	0	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	0	0



0	1	0	0	1	1	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	1	1	0	0	0
0	1	0	0	0	0	0	0	0	0	0	1	1	0
0	0	0	1	0	1	1	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	1	0	1	0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	1	0	0	0	0	0	1	1	0
0	0	0	0	0	0	1	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	1
1	0	0	0	0	1	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	1	1	0
0	0	0	0	0	1	1	1	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	1	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	1	0
0	1	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	1	0	0	0	1	0	0	0	0	1
0	0	1	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	1	0	0	0	0	0

0	1	0	1	0	1	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0	1	1	0	0	0
0	1	0	0	0	1	1	0	0	0	0	0	1	0
0	0	0	0	0	0	1	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	1	0	1	0
0	0	1	0	0	0	0	0	0	0	0	1	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	0	1
0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	1	1	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	1	0	1	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	0	1	0	0	0

0	0	0	0	0	0	0	0	0	0	0	1	1	0
0	0	1	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	1	1	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1
0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	1	0	1	0	0
0	0	0	0	1	0	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	1	0	0	0
0	0	0	0	1	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0	0	0	1	0	0	0
0	0	1	1	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	1	0	1	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	1	0	0	1	1	0	0	1
0	0	1	0	0	0	0	0	1	0	0	0	1	0
1	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	1	0	0	0	1	0	0	0	0	0

0	0	0	0	0	0	0	0	0	1	0	0	1	0
0	1	0	0	0	0	0	0	0	1	0	0	0	1
1	0	0	0	1	0	1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1	0	0	1	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	1	1	0	0	0
1	0	0	1	0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	1
1	0	0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	1	0	0	0	1	0	0	0	0	0
0	1	0	0	0	1	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	1
0	1	0	0	0	0	1	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	1
1	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	1	1	0	0	0
0	0	0	0	0	1	1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	1	0	0	0	0	0	0
1	0	0	0	0	0	1	0	0	1	1	0	0	0

0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	1	0	0	0	0	1	0	1
0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0
1	1	0	0	0	0	1	0	0	0	0	0	0	1
1	1	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	1	1	0	0
0	0	1	0	0	1	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	1	0	0	0	1	0	0	0	0	0	0
0	0	1	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1	0	1	0	0	0
1	1	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0
1	1	0	1	0	0	0	1	0	0	0	0	0	0
1	0	1	0	0	0	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	1	1	0	0	0	0	0	0	0	0

1	0	0	0	0	0	0	0	0	0	1	0	1	0
0	0	0	0	0	0	0	0	1	0	0	0	1	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	0
0	0	0	0	0	0	1	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	0	1	0	0	0	0	0
0	0	1	0	0	0	0	0	1	0	0	1	0	0
1	0	0	0	1	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	1	1	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	1	0	1	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	1	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	1	0	0
0	1	0	1	0	0	0	0	0	0	0	1	0	0
0	0	0	1	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	1

0	0	0	1	0	0	1	0	1	1	0	0	0	0
0	0	0	0	0	0	1	0	0	1	0	0	0	1
0	1	0	0	1	0	0	1	1	0	0	0	0	0
0	0	0	0	0	1	0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	1	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0	1	1	0
0	1	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	1	0	1	0	0
0	0	1	1	0	0	0	0	0	0	0	0	1	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	1	0	0	0
1	1	0	0	0	0	1	0	0	0	0	1	0	0
0	0	0	0	0	0	1	0	1	0	1	0	0	0
1	0	0	0	0	1	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	1	1	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	1	0	0	1	0	0	0	0	1

0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	0	1	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	1	0	0	1	0	1	0	1	0	0
0	0	0	0	0	0	1	0	1	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	1	0	0	1	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	1	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1	0	0	0	0	0



0	1	0	0	1	0	0	1	0	1	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	1	1	0
0	0	0	1	1	0	0	0	1	0	0	0	0	0
0	0	1	0	0	0	0	0	1	0	0	0	0	1
0	0	0	0	1	0	0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0	0	1	0	0	0	0
0	0	0	1	0	1	0	0	0	0	1	0	0	0
0	0	1	1	0	0	0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	1	0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	1

0	0	0	0	1	0	0	1	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	1	1	0	0	0
0	0	0	0	1	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0
1	0	0	0	0	0	0	0	0	1	0	0	1	0
0	0	1	1	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	1	1
0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1	1	0	0	0	0
0	0	1	0	1	0	0	1	0	0	0	1	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	0	0
0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0
0	0	0	0	0	0	0	1	0	0	0	0	1	1
0	0	1	0	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	0	0	1	0	0	1
0	0	0	0	1	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1	1	0	0	0	1	0

0	0	0	0	0	1	0	0	0	0	0	0	0	1
0	1	0	0	0	0	1	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	0	0	0	0	0	0
0	1	0	1	0	0	0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0	0	1	0	0	1	0
0	0	0	0	0	0	0	1	0	1	0	1	0	1
0	0	0	0	0	0	0	1	0	1	0	0	0	0
1	1	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	1	0	1
0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	1	0	0	1	0	1	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1	0	0	0	0	1	0
1	0	1	0	0	0	0	0	0	0	1	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0
1	0	1	0	1	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1	0	0	0	0	1	0

0	1	0	0	0	0	0	0	0	0	0	1	1	0
0	0	1	0	0	0	1	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	1	0
1	0	1	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	0	0	1	1	0
0	0	1	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0	0	0	0	0	1	1
0	0	0	1	0	0	0	1	0	0	0	0	1	0
0	1	0	1	0	0	0	0	0	0	0	1	0	0
0	1	0	1	0	0	0	0	0	1	0	0	0	1
0	1	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0	0	1	0	0	0	0
0	0	0	0	1	0	1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	1	1	0	0	0	0	0
0	0	0	1	0	0	0	0	0	1	0	0	0	0
1	1	0	1	0	0	0	0	0	0	0	0	0	0
1	0	0	1	1	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	1	0	0	1	0	0
0	1	0	0	1	0	0	0	0	0	1	0	0	1

0	0	0	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	0	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	1	0	1	0
0	0	0	0	0	0	0	0	1	0	0	0	0	1
0	0	0	1	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	1	0	1	1	0	0	0	0
1	0	1	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	1
1	0	0	1	0	0	0	1	0	0	0	1	0	0
1	0	0	0	1	1	0	0	0	0	0	1	0	0
0	1	0	0	0	0	0	0	0	1	1	0	0	0
0	0	0	0	0	1	0	0	0	0	1	0	1	0
0	0	0	0	1	1	0	0	0	0	0	0	0	1
0	0	0	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	1	0	0	0
1	1	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	1	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0

1	0	0	0	0	0	1	1	0	0	1	0	0	0
0	1	0	0	0	1	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	1	1	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	1	0	0	0
0	0	0	0	0	1	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	1	0	0	0	0	1	1	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	1	0	0	0
0	0	0	1	0	0	1	0	0	0	1	1	0	0
0	0	0	0	0	0	0	0	0	1	0	0	1	0
0	0	0	0	1	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	1	0	0	0	0	0	1	1
1	0	0	0	0	0	1	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0
1	0	0	0	1	1	0	0	0	0	0	0	0	0

0

Columns 15 through 24

0	0	1	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	1	0	0
0	0	0	1	0	0	1	0	0	0
0	0	1	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0
0	0	1	1	0	0	0	0	0	0
0	0	1	0	0	1	0	0	0	1
0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0
1	0	0	1	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	1	0	1	0	0
0	1	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	1	1	0	0
0	0	0	1	0	0	0	0	0	0
0	0	0	1	0	0	0	1	0	0
0	0	0	1	0	0	0	0	1	1
1	0	0	0	0	0	0	1	1	0
0	1	1	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	1	0
0	0	0	0	0	0	0	0	0	1
1	1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	1	0	0	0
0	1	1	0	1	0	0	1	0	0
0	0	1	0	1	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	1	0
1	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0
0	1	0	0	0	1	0	0	0	0
0	0	0	0	1	0	1	0	1	0
0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	1	0	1	0	0	0
0	1	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	1	0	1
0	0	0	1	0	0	0	1	0	1
1	0	1	0	0	0	0	0	0	1
1	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	1	1	1	0	0
0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	1	0	0	0
0	1	0	1	0	0	0	0	1	0
0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	0	1	0
1	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	1	1	0	0	0



0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	1	0	1	0	1
0	0	1	0	0	0	0	1	0	0
1	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	1	0	0	0
1	0	0	0	1	0	0	0	0	0
1	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1
1	0	0	1	0	0	0	1	1	0
0	0	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	0	0
1	0	1	0	0	1	0	0	1	0
0	0	1	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	1	0	0	0
0	1	0	0	0	1	0	1	0	0
0	0	0	1	0	0	0	0	0	0
0	1	1	0	0	0	0	1	0	0
0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	0	1	0
0	0	0	0	1	0	0	1	0	0
0	0	1	0	1	0	0	0	1	0
0	0	0	1	0	1	0	0	1	0
0	0	0	0	1	0	1	0	0	0
0	0	1	1	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0
1	0	0	0	0	0	0	1	1	0
0	0	0	0	0	0	0	0	1	0
1	0	1	0	0	0	0	0	0	0
0	1	0	0	0	0	1	0	1	0
0	0	1	0	0	0	1	1	0	0
1	1	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	1
0	0	0	0	1	0	0	0	1	0
0	0	1	0	0	0	0	1	0	0

0	0	1	1	0	0	0	0	1	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0
1	0	0	0	0	0	0	0	1	0
0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	1	0	1
0	0	0	1	0	0	1	0	0	0
0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0
0	1	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0
0	0	1	0	0	0	0	1	0	1
0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0
0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	1	0	0
0	1	0	0	0	0	0	0	0	1
1	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	1	0	0
0	0	0	1	0	0	0	1	0	0
1	0	0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0
0	1	0	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	1	0
0	0	1	0	0	0	0	0	0	1
0	0	1	0	0	1	1	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0
1	0	0	0	0	1	1	0	0	0
0	1	0	0	0	0	0	0	0	0
0	0	0	0	1	1	0	1	0	0
1	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0
0	0	0	0	1	0	1	0	0	1
0	0	0	0	0	0	1	0	0	1
0	0	0	0	1	0	1	0	0	0
0	0	1	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

1	0	0	0	0	0	0	0	1	0
1	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0
0	0	1	0	1	0	1	0	0	0
0	0	0	0	0	0	1	0	0	1
0	0	1	0	0	0	0	1	0	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	1	0
1	0	0	0	1	0	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	1	1	0	0	0	0	0	1	0
1	0	0	0	1	0	0	0	0	0
0	0	1	0	0	0	1	0	0	0
0	0	0	0	1	0	1	0	0	0
0	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	1	0	0	1
0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	1	0	0	0
1	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	1	0
0	1	0	0	1	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
0	1	0	0	1	0	0	0	1	0
0	0	0	1	1	0	0	0	0	0
0	0	1	0	0	0	1	0	0	0
0	0	0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	1	0	0	1	0	0
0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0
0	1	0	1	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	1	0

0	0	1	0	1	0	0	0	0	1
1	0	1	0	1	0	1	0	0	0
0	0	0	0	0	1	0	0	0	0
0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	1	0	0	0
0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	1	1
0	0	1	1	0	0	0	0	1	0
0	0	0	1	1	1	0	0	0	0
0	0	0	0	1	0	0	0	1	1
1	0	0	0	1	0	0	0	0	1
0	0	1	0	0	1	0	0	0	0
0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0
0	1	0	1	0	0	0	0	0	0
1	1	0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	1	0	0	1	0	0	0
1	1	0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	0	1	0	1	0
0	0	0	0	1	1	0	0	0	0
0	0	0	1	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	1
0	1	1	0	0	0	0	0	1	1
0	0	0	1	0	0	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	1	0	0	1
1	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0
1	0	0	0	0	0	0	0	0	1
0	1	1	0	0	0	0	0	0	0
0	0	0	0	1	1	0	1	0	0
1	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	1	0	0	0

0	0	0	1	1	0	0	0	0	0
1	0	0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	1	0	0
0	0	0	1	1	0	0	0	0	0
0	0	1	0	1	0	0	0	0	1
0	0	0	0	1	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	1	1	0	0	0
1	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	1	0	1	0	0
1	0	0	1	0	0	0	0	0	0
0	1	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	1	0
1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	1
0	1	1	1	1	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	1	0	0
0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0
0	1	0	0	1	0	1	0	0	0
0	1	0	0	1	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	1	1	0	0
0	1	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0
0	0	1	0	0	1	0	0	0	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	1	0	0
1	0	0	0	0	1	0	0	0	0
0	0	0	1	1	0	0	1	0	0
0	0	1	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	1	1	0
0	0	0	0	0	0	1	0	0	0
0	1	0	0	1	0	1	0	0	0

1	0	0	0	0	0	0	1	1	0
1	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	1	1	0	0	0
0	0	0	0	1	0	0	1	0	1
1	1	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	0	0	1
0	0	0	0	0	0	0	1	0	0
0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	1	0	0
0	0	0	0	0	0	1	0	0	0
1	0	0	1	0	0	1	0	0	0
0	0	0	1	0	0	0	0	1	0
0	0	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	1	0
1	0	1	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	1
0	0	0	0	0	0	0	0	0	1
0	1	0	0	0	0	1	0	0	0
0	1	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	1	0	0	0
0	0	0	0	1	1	0	0	0	0
0	1	0	0	0	0	0	0	0	1
1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	1
0	0	0	0	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	1	0	0	0
0	0	0	1	1	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	1	1
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0

0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	1	0	0
0	0	0	1	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	0
0	0	0	1	0	0	0	0	0	1
0	1	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	1	1	0
0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0	1	0
0	0	0	0	0	1	0	0	0	1
0	0	0	1	1	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	1	0
0	0	1	1	0	1	0	0	1	0
0	0	0	0	1	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0